REPORT DOCUMENTATION PAGE

Form Approved OMB NO. 0704-0188

1. REPORT DATE (DD-MM-YYYY)	2. REPORT TYPE	3. DATES COVERED (From - To)
23-10-2011	Conference Proceeding	-

4. TITLE AND SUBTITLE
Polytope Codes Against Adversaries in Networks

5a. CONTRACT NUMBER
W911NF-10-1-0419

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER
611102

6. AUTHORS
Oliver Kosut, Lang Tong, David Tse

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAMES AND ADDRESSES
Cornell University
Office of Sponsored Programs
Cornell University
Ithaca, NY 14853 -2801

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
U.S. Army Research Office
P.O. Box 12211
Research Triangle Park, NC 27709-2211

10. SPONSOR/MONITOR'S ACRONYM(S)
ARO

11. SPONSOR/MONITOR'S REPORT NUMBER(S)
58094-NS.2

12. DISTRIBUTION AVAILIBILITY STATEMENT
Approved for public release; distribution is unlimited.

13. SUPPLEMENTARY NOTES
The views, opinions and/or findings contained in this report are those of the author(s) and should not contrued as an official Department of the Army position, policy or decision, unless so designated by other documentation.

14. ABSTRACT
Network coding is studied when an unknown subset of nodes in the network is controlled by an adversary. To solve this problem, a new class of codes called Polytope Codes is introduced. Polytope Codes are linear codes operating over bounded polytopes in real vector fields. The polytope structure creates additional complexity, but it induces properties on marginal distributions of code vectors so that validities of codewords can be checked by internal nodes of the network. It is shown that a cut-set bound for a class planar networks can be achieved using Polytope

15. SUBJECT TERMS
Network Coding. Byzantine Attack. NetworkError Correction. Nonlinear Codes.

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	15. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			Lang Tong
UU	UU	UU	UU		19b. TELEPHONE NUMBER: 607-255-3900

Report Title

Polytope Codes Against Adversaries in Networks

ABSTRACT

Network coding is studied when an unknown subset of nodes in the network is controlled by an adversary. To solve this problem, a new class of codes called Polytope Codes is introduced. Polytope Codes are linear codes operating over bounded polytopes in real vector fields. The polytope structure creates additional complexity, but it induces properties on marginal distributions of code vectors so that validities of codewords can be checked by internal nodes of the network. It is shown that a cut-set bound for a class planar networks can be achieved using Polytope Codes. It is also shown that this cut-set bound is not always tight, and a tighter bound is given for an example network.

Conference Name: 2010 Intl Symp. Inform. Theory

Conference Date:

Polytope Codes Against Adversaries in Networks

Oliver Kosut, Lang Tong, and David Tse

***Abstract*— Network coding is studied when an unknown subset of nodes in the network is controlled by an adversary. To solve this problem, the class of polytope codes is introduced. Polytope codes are linear codes operating over bounded polytopes in real vector fields. The polytope structure creates additional complexity, but it induces properties on marginal distributions of code vectors so that validities of codewords can be checked by internal nodes of the network. It is shown that a cut-set bound for a class planar networks can be achieved by the polytope codes. It is also shown that this cut-set bound is not always tight, and a tighter bound is given for an example network.**

I. Introduction

Network coding allows routers in a network to execute possibly complex codes in addition to mere forwarding; it has been shown that allowing them to do so can increase throughput [1]. However, taking advantage of this use of coding at internal nodes means that the sources and destinations must rely on other nodes—nodes they may not have complete control over—to reliably perform certain functions. If these internal nodes do not perform the function correctly, or, worse, maliciously attempt to subvert the goals of the users, launching a so-called Byzantine attack [2], [3], standard network coding techniques fail.

Suppose an omniscient adversary controls an unknown portion of the network, and may arbitrarily corrupt the values sent on certain links. We wish to determine how the size of the adversarial part of the network influences the capacity. If the adversary may control any z unit-capacity edges in the network, then it has been shown that, for the multicast problem (one source and many destinations), the capacity reduces by $2z$ compared to the non-Byzantine problem [4], [5]. To achieve this rate, only linear network coding is needed. Furthermore, if there is just one source and one destination, only routing is needed at internal nodes.

The above model assumes that any set of z edges may be adversarial, which may be overly pessimistic depending on the situation. If the adversary cuts a certain number of transmission lines in a network, this would be a reasonable model. If, on the other hand, the adversary seizes a single router, it will control the values on all links connected to that router; the number of these links may vary in number depending on which router is attacked. It is easy to construct examples (see [6]) for which allowing the adversary to control any set of z edges results in a much lower throughput than allowing them only to control z edges if they all emerge from the same node. It is therefore reasonable to consider the problem of the adversary controlling any set of z nodes, as we do in this paper.

O. Kosut and L. Tong are with Cornell University, Ithaca, NY {oek2,lt35}@cornell.edu

D. Tse is with the University of California, Berkeley, CA dtse@eecs.berkeley.edu

This work is supported in part by the National Science Foundation under Award CCF-0635070 and the Army Research Office under Grant ARO-W911NF-10-10419.

A. Related Work

Byzantine attacks on network coding were first studied in [7], which looked at detecting adversaries in a random linear coding environment. The z unit-capacity edge adversary problem was solved theoretically in [4], [5]. In [8], the same problem is studied, providing distributed and low complexity coding algorithms to achieve the same asymptotically optimal rates. In addition, [8] looks at two adversary models slightly different from the omniscient one considered in [4], [5] and in this paper. They show that higher rates can be achieved under these alternate models. In [9], a more general view of the adversary problem is given, whereby the network itself is abstracted into an arbitrary linear transformation.

Network coding under Byzantine attacks that are more general than the simple edge-based model was first studied in [6] and [12]. The former studied node-based attacks by means of several examples, and the latter looked at the problem of edge-based attacks when the edges could have unequal capacities. This problem was found to have similar complications to the node-based problem. Both found that linear coding is suboptimal, and that simple nonlinear operations used to augment a linear code can improve throughput. It is shown in [13] that the node-based problems subsumes even the unequal edge problem.

All of these works, in addition to ours, seek to correct for the adversarial errors at the destination. An alternative strategy known as the watchdog, studied for wireless network coding in [10], is for nodes to police downstream nodes by overhearing their messages to detect modifications. In [11], a similar approach is taken, and they found that nonlinear operations similar to ours can be helpful, in which comparisons are made to detect errors.

B. Main Results

The primary contribution of the present paper is to elaborate the theory of polytope codes, originally introduced in [6] under the less descriptive term “bounded-linear codes”. Polytope codes make use of probability distributions defined over polytopes in real vector fields. The main innovation of these distributions is that it they possess properties having to do with constraints against their marginal distributions. This property is stated and proved is Section V, and it allows more effective checks at internal nodes in a network; these checks either detect and correct adverserial actions or force adversaries to act properly.

Polytope codes are used to prove that the cut-set bound, stated in Section III, is tight for a certain class of planar networks. Planarity requires that the graph can be embedded in a plane such that intersections between edges occur at nodes. This allows additional comparisons that might otherwise not be present, allowing the code to more well defeat Byzantine attacks.

Finally, we show in Section VIII that the cut-set bound is not always tight, by giving an example with a tighter bound.

II. Problem Formulation

Let (V, E) be an directed acyclic graph. For each edge $e \in E$, there is an edge capacity c_e, which we assume to be an integer. One node in V is denoted S, the source, and one is denoted D, the destination. We wish to determine the maximum achievable throughput from S to D when any set of z nodes in $V \setminus \{S, D\}$ are *traitors*; i.e. they are controlled by the adversary. Given a rate R and a block-length n, the message W is chosen at random from the set $\{1, \ldots, 2^{nR}\}$. Each edge e holds a value $X_e \in \{1, \ldots, 2^{nc_e}\}$.

A code is be made up of three components:

1) an encoding function at the source, which produces values to place on all the output edges given the message,
2) a coding function at each internal node $i \in V \setminus \{S, D\}$, which produces values to place on all output edges from i given the values on all input edges to i,
3) and a decoding function at the destination, which produces an estimate $\hat{W}$ of the message given the values on all input edges.

Suppose $T \subseteq V \setminus \{S, D\}$ with $|T| = z$ is the set of traitors. They may subvert the coding functions at nodes $i \in T$ by placing arbitrary values on all the output edges from these nodes. Let Z_T be the set of values on these edges. For a particular code, specifying the message W as well as Z_T determines exactly the values on all edges in the network, in addition to the destination's estimate $\hat{W}$. We say that a rate R is *achievable* if there exists a code operating at that rate with some block-length n such that for all messages, all sets of traitors T, and all values of Z_T, $W = \hat{W}$. That is, the destination always decodes correctly no matter what the adversary does. Let the *capacity* C be the supremum over all achievable rates.

III. Cut-Set Upper Bound

Theorem 1: Consider a cut $A \subseteq V$ with $S \in A$ and $D \notin A$. Let E_A be the set of edges that cross the cut. For two not necessarily disjoint sets of possible traitors T_1, T_2, let E_1 and E_2 be the subset of edges in E_A that originate at nodes in T_1 and T_2 respectively. Let $\tilde{E}$ be the set of edges in $E_1 \cap E_2$ in addition to all edges $e \in E_1 \cup E_2$ for which there is no path that flows through e followed by any edge in $E_A \setminus E_1 \setminus E_2$. The following upper bound holds on the capacity of the network:

$$C \leq \sum_{e \in E_A \setminus \tilde{E}} c_e. \tag{1}$$

Proof: A version of this theorem was proved in [6]. The proof follows along the lines of the Singleton bound. ∎

The corresponding cut-set bound when an arbitrary z unit-capacity edges can be modified by the adversary was proved in [4]. In [12], it was conjectured (for the edge adversary problem with unequal capacities) that the cut-set bound (stated more in the more general form than that in [4], but essentially the same as Theorem 1) is not tight. We prove in Sec. VIII that the cut-set bound stated in Theorem 1 is not tight in general. The example used to demonstrate this, though it is a node adversary problem, can be easily modified to confirm the conjecture stated in [12].

IV. Capacity of A Class of Planar Networks

Theorem 2: Let (V, E) be a network with the following properties:

1) It is planar.
2) No node other than the destination has more than two unit-capacity input edges (i.e. either one 2-capacity edge or two unit-capacity edges).
3) No node other than the source has more output capacity than input capacity.

If $z = 1$, cut-set bound is tight for this network.

Proof: Omitted due to space limitation. See [13]. Section VI illustrates the proof for an example, and Section VII briefly sketches how planarity is used. ∎

The key ingredient in the proof of the above theorem is the use of a new class of codes referred to as Polytope Codes, introduced next in Sec. V. Polytope codes are nonlinear codes, but they are not drastically different from linear codes; they are linear codes defined on polytopes in real field. The proof of the above theorem seems to suggest that the cut-set bound may be tight for a much larger class of networks. Indeed, we conjecture that this theorem can be generalized, and that polytope codes achieve capacity for all planar networks and all z.

Fig. 1 shows an example of a network that satisfies the conditions of Theorem 2. Thus the capacity of this network (achieved by a polytope code) is 4. If only linear codes are allowed, the maximum achievable rate is 3 (see [6] for a proof of a similar fact). The polytope code used to prove achievability is discussed in detail in Section VI.

V. The Polytope Code

We begin with a simple example of the distribution underlying the polytope code. For some positive integer k, consider the set of $X, Y, Z, W \in \{-k, \ldots, k\}$ satisfying

$$X + Y + Z = 0 \tag{2}$$

$$3X - Y + 2W = 0. \tag{3}$$

This is the set of integer lattice points in a polytope. Let the distribution $p(xyzw)$ be uniform over these points. The region of (X, Y) pairs with positive probability is shown in Fig. 2. Observe that even though X and Y are linearly independent in the subspace given by (2)–(3), they are not statistically independent, because the boundedness of Z and W requires that X and Y satisfy certain linear inequalities. Nevertheless,

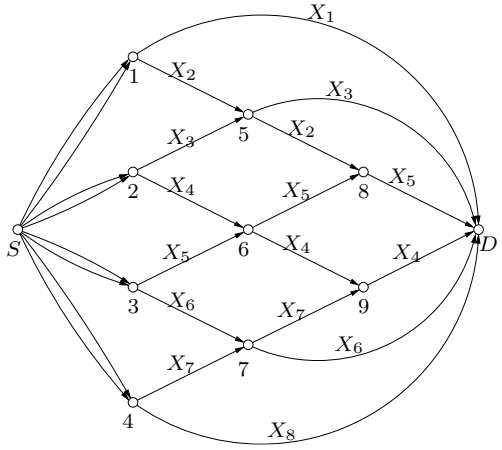

Fig. 1. An example planar network. All nodes may be traitors, and all edges have capacity 1. The specification of a capacity-achieving polytope code is given.

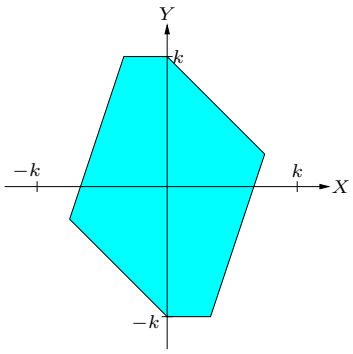

Fig. 2. An example polytope projected into the (X, Y) plane.

the area of the polygon shown in Fig. 2 grows as $\mathcal{O}(k^2)$. Hence,

$$\lim_{k\to\infty} \frac{H(XY)}{\log k} = 2. \tag{4}$$

Note also that

$$\lim_{k\to\infty} \frac{H(X)}{\log k} = \lim_{k\to\infty} \frac{H(Y)}{\log k} = 1. \tag{5}$$

Therefore, for large k, X and Y are nearly independent in that their joint entropy is close to the sum of the individual entropies. The four variables X, Y, Z, W make up something like a $(4, 2)$ MDS code, in that each pair is close to independent for large k, and any two completely determine the other two. These distributions over polytopes can be made to perform many of the same functions as standard linear variables defined over finite fields.

More generally, given a matrix $F \in \mathbb{Z}^{u\times m}$, consider the polytope

$$\left\{x \in \mathbb{Z}^m : Fx = 0, |x_i| \le k \text{ for } i = 1, \ldots, m\right\}. \tag{6}$$

We may also describe this polytope in terms of a matrix K whose columns form a basis for the null-space of F. Let $p(x)$ be a uniform distribution over this polytope. In a polytope code, each codeword is a sequence x^n with joint type equal to $p(x)$. Each edge in the network holds a sequence x_i^n. The following lemma generalizes the entropy calculations performed above.

Lemma 1: According to $p(x)$, for any $S \subseteq \{1, \ldots, m\}$

$$\lim_{k\to\infty} \frac{H(X_S)}{\log k} = \text{rank}(K_S) \tag{7}$$

where K_S is the matrix made up of the rows of K corresponding to the elements of S.

Recall that in a linear code operating over the finite field $\mathbb{F}$, we may express the elements on the edges in a network $x \in \mathbb{F}^m$ as linear combinations of the message $x = Kw$, where K is a linear transformation over the finite field, and w is the message vector. Taking a uniform distribution on w imposes a distribution on X such that $H(X_S) = \text{rank}(K_S) \log |\mathbb{F}|$. This differs from (7) only by a constant factor, and also that (7) holds only in the limit of large k. Hence, polytope codes achieve a similar set of entropy profiles as linear codes.

In a polytope code, every node in the network observes several sequences x_i^n, from which it may determine their joint type. It will check whether the joint type matches the corresponding distribution from p, and forward a bit specifying whether it does along all its outgoing edges. In addition, all nodes will continue forwarding these comparison bits all the way to the destination. The number of these bits is determined only by the size of the network, so they occupy an arbitrarily small amount of link capacity as n grows. These comparisons force a traitor to make a choice: it causes the comparison to fail, which may give away its location, or it is constrained so that the comparison pass. The following theorem allows us to analyze how these constraints influence the traitors' actions.

Theorem 3 (Fundamental Property of Polytope Codes): Let $p(x)$ be a probability distribution for $x \in \{-k, \ldots, k\}^m$ such that $p(x) > 0$ only if $Fx = 0$. For another distribution $q(x)$ on the same space subject to

$$q(A_l x) = p(A_l x) \text{ for } l = 1, \ldots, L \tag{8}$$

where $A_l \in \mathbb{R}^{u_l \times m}$, q must be identical to p if the following properties of F and the A_l hold:

1) There exists a positive definite matrix C such that

$$F^T C F = \sum_{l=1}^{L} A_l^T \Sigma_l A_l \tag{9}$$

for some $\Sigma_l \in \mathbb{R}^{u_l \times u_l}$.

2) There exists an l^* such that the value of $A_{l^*}x$ uniquely determines x subject to $Fx = 0$. That is, the matrix $\begin{bmatrix} A_{l^*} \\ F \end{bmatrix}$ has full column rank.

Proof: By (8), for all l and all $y \in \{-k, \ldots, k\}^{u_l}$,

$$\sum_{x:A_l x = y} q(x) = \sum_{x:A_l x = y, Fx = 0} p(x) \tag{10}$$

where we have used the fact that $p(x)$ is only positive if $Fx = 0$. Multiplying (10) by $y^T \Sigma_l y$ and summing over all y and all l, then applying (9), yields

$$\sum_x [x^T F^T C F x] q(x) = \sum_{x:Fx=0} [x^T F^T C F x] p(x). \tag{11}$$

Observe that $x^T F^T C F x$ is 0 if $Fx = 0$ and positive if $Fx \neq 0$, because C is positive definite. Therefore the right hand side of (11) is 0, and the left hand side is a linear combination of $\{q(x) : Fx \neq 0\}$ with strictly positive coefficients. Therefore $q(x) = 0$ if $Fx \neq 0$.

Now we make use of the property (2). For any $x \in \{-k, \ldots, k\}^m$ with $Fx = 0$, we may rewrite (10) for l^* as

$$\sum_{x' : A_{l^*} x' = A_{l^*} x, Fx' = 0} q(x') = \sum_{x' : A_{l^*} x' = A_{l^*} x, Fx' = 0} p(x'). \quad (12)$$

But observe that $A_{l^*} x' = A_{l^*} x$ and $Fx' = 0$ imply that $x' = x$. That is, there is only one term in the summation. Hence, $q(x) = p(x)$ for all x. ∎

VI. Application of Polytope Codes Against Adversary Nodes

Consider the example in Fig. 1, shown with a routing scheme for a polytope code that we will show achieves the cut-set bound of 4 with one traitor. In Fig. 1, variables X_1–X_8 comprise the polytope code equivalent of an $(8, 4)$ MDS code. That is, the matrix F imposing constraints on the Xs is such that any four variables are linearly independent, and determine the other four. Recall that each of these variables represents a sequence x_i^n transmitted across the respective link so that $(x_1^n, \ldots, x_8^n)$ has joint type equal to $p(x_1 \cdots x_8)$, and each message is associated with one such joint sequence. In addition to these sequences, comparison bits are generated and sent through the network, as discussed above.

To decode, the destination first compiles a list $\mathcal{L} \subset V$ of which nodes may be the traitor. It does this by taking all its available data: received comparison bits as well as the x_i^n sequences it has access to, and determines whether it is possible for each node, if it were the traitor, to have acted in a way to cause these data to occur. If so, it adds that node to $\mathcal{L}$. For example, if the comparison bit from node 8 reports that the distribution of X_2 and X_5 did not match p, then the traitor must be a node that can influence one of those two symbols; i.e. $\mathcal{L} \subset \{1, 3, 5, 6, 8\}$. Observe that the true traitor is in $\mathcal{L}$.

Depending on $\mathcal{L}$, the destination choses which variables to decode from. Any traitor action leads to some $\mathcal{L}$, so it is comprehensive to consider all values for $\mathcal{L}$. If $|\mathcal{L}| = 1$, then this single node must be the traitor, so the destination can simply disregard all symbols that came into contact with that node (there are at most 2) and decode the message from the rest. Now consider the case that $|\mathcal{L}| \geq 2$. Say node i is the traitor, and $j \in \mathcal{L}$, $j \neq i$. This places certain constraints on the behavior of node i that we may exploit. The following lemma allows us to conclude that the code is effective.

Lemma 2: If node i is the traitor, but $j \in \mathcal{L}$, then i could only have corrupted variables that are also touched by node j.

By Lemma 2, the destination can ignore variables touched by all nodes in $\mathcal{L}$ and decode from the rest. For example, suppose node 3 is the traitor and node 6 is in $\mathcal{L}$. The only symbol node 3 may have corrupted is X_5, so the destination may decode from the rest.

To prove Lemma 2, in principle one needs to check the condition for all pairs i and j. We illustrate the argument for one pair; generalization is not hard. Suppose that node 3 is the traitor and it acts such that it appears to the destination that node 2 may be the traitor. Because node 2 and node 3 do not share any symbols, we wish to show that node 3 cannot transmit to nodes 6 and 7 anything but the true values of X_5 and X_6 respectively. Let $q(x_1 \cdots x_8)$ be the empirical type of the sequences sent through the network, where it may be different from p if node 3 has changed X_5 or X_6. The following conditions hold on q:

$$q(x_1 x_2 x_7 x_8) = p(x_1 x_2 x_7 x_8), \quad (13)$$

$$q(x_2 x_5) = p(x_2 x_5), \quad (14)$$

$$q(x_6 x_7) = p(x_6 x_7), \quad (15)$$

$$q(x_1 x_5 x_6 x_8) = p(x_1 x_5 x_6 x_8). \quad (16)$$

Condition (13) holds because X_1, X_2, X_7, X_8 cannot be influenced by node 3, so they retain their true values. Condition (14) holds because if it did not, node 5, which observes X_2 and X_5, would detect it, and forward the information to the destination. Since node 2 cannot influence X_2 or X_5, this would remove 2 from $\mathcal{L}$. Similarly, (15) holds because X_6, X_7 are compared at node 7. Finally, if (16) did not hold, then the destination could tell that node 2 was not the traitor, because X_1, X_5, X_6, X_8 are untouched by node 2.

We wish now to apply (13)–(16) to Theorem 3 to conclude

$$q(x_1 x_2 x_5 x_6 x_7 x_8) = p(x_1 x_2 x_5 x_6 x_7 x_8). \quad (17)$$

This is enough to show that X_5 and X_6 cannot be corrupted by node 3, because under p, X_1, X_2, X_7, X_8 determine X_5 and X_6, so they must do so in exactly the same way under q. We must now check that (13)–(16) satisfy the conditions of Theorem 3. The second condition can be satisfied by choosing l^* to correspond to either (13) or (16), since they both involve four variables, and by construction any four variables determine the rest. Satisfying the first condition is not so easy: there exist Σ_l matrices such that the right hand side of (8) can be made to be any matrix whose entries corresponding to the pairs (X_2, X_6) and (X_5, X_7) are zero, because these pairs never exist simultaneously in any of (13)–(16). This places two linear constraints C. Observe that C is 2×2, since the six variables are subject to two linear constraints. It can be shown that there exists a positive definite C iff

$$|K_{1,2,7,8}| \; |K_{1,5,6,8}| \; |K_{1,2,5,8}| \; |K_{1,7,6,8}| < 0 \quad (18)$$

Observe that in order for X_1–X_8 to be the equivalent of an $(8, 4)$ MDS code, each of the determinants in (18) must be nonzero; here we see that we must design the code being aware of the signs of these determinants as well as that they be nonzero. It is, however, possible to design K to satisfy (18). This concludes our proof of Lemma 2 for $i = 3$ and $j = 2$.

VII. Planar Networks

We briefly sketch how the arguments in Section VI can be extended to prove Theorem 2. We need to generate a routing scheme for an arbitrary network, then prove a more general form of Lemma 2. The key observation in order to do this in general is that the important comparisons that go on inside the

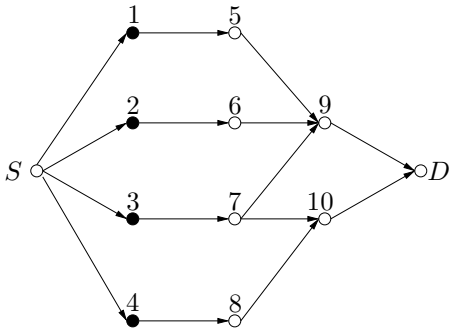

Fig. 3. A network with capacity strictly less than the cut-set bound.

network are those that involve a variable that does not reach the destination (e.g. (14) and (15) involve X_2 and X_7). This is because those symbols that do reach the destination can be examined there, so further comparisons inside the network do not add anything. Therefore the key routing problem is to carefully design the paths of these non-destination symbols to maximize the utility of their comparisons. In particular, we design these paths so that, as much as possible, for a node having one direct edge to the destination and one other output edge, the output edge not going to the destination holds a non-destination variable. The advantage of this is that any variable, before exiting the network, is guaranteed to cross a non-destination variable at a node where the two variables may be compared, but this routing requirement may not be possible if the network is not planar.

VIII. Looseness of the Cut-set Bound

We show that the cut-set bound given in Theorem 1 is not tight. We do this in two parts. First, consider the problem that a special subset of nodes are designated as potential traitors, and the code must guard against adversarial control of any z of those nodes. We refer to this as the limited-node problem. Certainly the limited-node problem subsumes the all-node problem, since we may simply take the set of potential traitors to be all nodes. Furthermore, it subsumes the unequal-edges problem studied in [12], because given an instance of the unequal-edge problem, an equivalent limited-node problem can be constructed as follows: create a new network with every edge replaced by a pair of edges of equal capacity with a node between them. Then limit the traitors to be only these interior nodes. It can be shown (see [13]) that the all-node problem actually subsumes the limited-node problem. This is done by constructing all-node problems equivalent to a limited-node problem. Next we give an an example of a limited-node network for which there is an active upper bound on capacity tighter than the cut-set. This proves that, even for the all-node problem, the cut-set bound is not tight.

Consider the network shown in Figure VIII. All edges have capacity one, and there is at most one traitor, but it is restricted to be one of the black nodes. The cut-set bound is 2, but in fact the capacity is no more than 1.5.

Consider a code achieving rate R. For $i = 1, 2, 3, 4$, let X_i be the random variable representing the value on the output edge of node i. Let Y be the value on edge $(9, D)$ and let Z be the value on $(10, D)$. Let p be the honest distribution on these variables, and define the following alternative distributions:

$$q_3 = p(x_1x_2x_4)p(x_3)p(y|x_1x_2x_3)p(z|x_3x_4), \quad (19)$$

$$q_4 = p(x_1x_2x_3)p(x_4)p(y|x_1x_2x_3)p(z|x_3x_4). \quad (20)$$

If node 3 or node 4 is the traitor, they may induce these distributions. Therefore

$$R \leq I_{q_3}(X_1X_2X_4; YZ), \quad (21)$$

$$R \leq I_{q_4}(X_1X_2X_3; YZ). \quad (22)$$

Observe that $q_3(x_3x_4z) = q_4(x_3x_4z)$, meaning any joint entropy made up of these three variables is the same for each distribution. Using this fact, (21), (22), that all edges have capacity 1, and standard information theoretic inequalities, one can conclude that $R \leq 1.5$.

IX. Conclusion

The main contribution of this paper has been to introduce the theory of polytope codes. As far as we know, they are the best known coding strategy to defeat generalized Byzantine attacks on network coding. However, it remains difficult to calculate the best possible rate they can achieve for a given network. We have proved that they achieve the cut-set bound, and hence the capacity, for a class of planar graphs, and we conjecture that this holds for all planar graphs. One would obviously hope to find the capacity of all networks, including non-planar ones. We have shown that achieving the cut-set bound is not always possible, meaning there remains significant work to do on upper bounds as well as achievable schemes. Whether polytope codes can achieve capacity on all networks remains an important open question.

References

[1] R. Ahlswede, N. Cai, S. R. Li, and R. W. Yeung, "Network information flow," *IEEE Trans. Inform. Theory*, vol. 46, pp. 1204–1216, 2000.

[2] L. Lamport, R. Shostak, and M. Pease, "The Byzantine generals problem," *ACM Transactions on Programming Languages and Systems*, vol. 4, pp. 382–401, 1982.

[3] D. Dolev, "The Byzantine generals strike again," *Journal of Algorithms*, vol. 3, no. 1, pp. 14–30, 1982.

[4] N. Cai and R. W. Yeung, "Network error correction, part I: Basic concepts and upper bounds," *Comm. in Inf. and Syst.*, vol. 6, no. 1, pp 19–36, 2006.

[5] N. Cai and R. W. Yeung, "Network error correction, part II: Lower bounds," *Comm. in Inf. and Syst.*, vol. 6, no. 1, pp 37–54, 2006.

[6] O. Kosut, L. Tong, and D. Tse, "Nonlinear network coding is necessary to combat general Byzantine attacks," in *Proc. Allerton*, Sept. 2009.

[7] T. Ho, B. Leong, R. Koetter, M. Médard, M. Effros, and D. R. Korger, "Byzantine modification detection in multicast networks using randomized network coding," in *Proc. ISIT*, 2004.

[8] S. Jaggi, et al, "Resilient network coding in the presence of Byzantine adversaries," in *Proc. INFOCOM*, pp. 616–624, 2007.

[9] R. Koetter, F. R. Kschischang, "Coding for Errors and Erasures in Random Network Coding," *IEEE Transactions on Information Theory*, vol. 54, no. 8, pp. 3579-3591.

[10] M. Kim, M. Médard, J. Barros, R. Koetter, "An algebraic watchdog for wireless network coding," in *Proc. ISIT*, June 2009.

[11] G. Liang and N. H. Vaidya, "When watchdog meets coding," *Technical Report*, May 2009.

[12] S. Kim, T. Ho, M. Effros, and S. Avestimehr, "Network error correction with unequal link capacities," in *Proc. Allerton*, Sept. 2009.

[13] O. Kosut, L. Tong, D. Tse, "Node-based Byzantine attacks on network coding", to be submitted to *IEEE Trans. Info. Theory*.